

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of:

Toshikazu YOSHIDA et al.

Application No.: Unassigned

Group Art Unit: Unassigned

Filed: April 1, 2004

Examiner: Unassigned

For: INTEGRATED CIRCUIT DEVICE HAVING SEND/RECEIVE MACRO FOR SERIAL  
TRANSFER BUS

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN  
APPLICATION IN ACCORDANCE  
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Commissioner for Patents  
PO Box 1450  
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s) herewith  
a certified copy of the following foreign application:

Japanese Patent Application No(s). 2003-363038


Filed: October 23, 2003

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing  
date(s) as evidenced by the certified papers attached hereto, in accordance with the  
requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: April 1, 2004

By:   
H. J. Staas  
Registration No. 22, 010

1201 New York Ave, N.W., Suite 700  
Washington, D.C. 20005  
Telephone: (202) 434-1500  
Facsimile: (202) 434-1501

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年10月23日  
Date of Application:

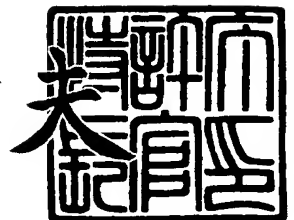
出願番号 特願2003-363038  
Application Number:  
[ST. 10/C]: [JP 2003-363038]

出願人 富士通株式会社  
Applicant(s):

2003年12月22日

特許庁長官  
Commissioner,  
Japan Patent Office

今井 康



出証番号 出証特2003-3106469

【書類名】 特許願  
【整理番号】 0340855  
【提出日】 平成15年10月23日  
【あて先】 特許庁長官 殿  
【国際特許分類】 G06F 13/00  
【発明者】  
    【住所又は居所】 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 富士通株式会社  
                                内  
    【氏名】 吉田 俊一  
【発明者】  
    【住所又は居所】 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 富士通株式会社  
                                内  
    【氏名】 谷 昌昭  
【発明者】  
    【住所又は居所】 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 富士通株式会社  
                                内  
    【氏名】 古屋 健二  
【発明者】  
    【住所又は居所】 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 富士通株式会社  
                                内  
    【氏名】 原 章雄  
【発明者】  
    【住所又は居所】 神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号 富士通株式会社  
                                内  
    【氏名】 倉田 智生  
【特許出願人】  
    【識別番号】 000005223  
    【氏名又は名称】 富士通株式会社  
【代理人】  
    【識別番号】 100094525  
    【弁理士】  
    【氏名又は名称】 土井 健二  
【選任した代理人】  
    【識別番号】 100094514  
    【弁理士】  
    【氏名又は名称】 林 恒徳  
【手数料の表示】  
    【予納台帳番号】 041380  
    【納付金額】 21,000円  
【提出物件の目録】  
    【物件名】 特許請求の範囲 1  
    【物件名】 明細書 1  
    【物件名】 図面 1  
    【物件名】 要約書 1  
    【包括委任状番号】 9704944

## 【書類名】特許請求の範囲

## 【請求項 1】

シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記所定単位のデータの送信に応答して受信側デバイスから送信されるデータアクノリッジ信号を検出するアクノリッジ信号検出ユニットと、

前記アクノリッジ信号検出ユニットによる前記データアクノリッジ信号の検出に応答して、前記CPUへの割込を発生することなく前記送受信バッファに格納されたデータを送信するデータ送信ユニットとを有し、

前記アクノリッジ検出ユニットは、前記所定単位のデータの送信に対応して受信側デバイスから送信されるデータアクノリッジ信号を検出しない場合に、前記CPUにデータアクノリッジ信号未検出の割込を発生することを特徴とする集積回路装置。

## 【請求項 2】

請求項 1 において、

前記アクノリッジ検出ユニットは、

前記データ送信ユニットが、マスターとして、スレーブデバイスを特定するアドレスを送信したことに応答して、前記スレーブデバイスから送信されるアドレスアクノリッジ信号を検出した場合に、前記CPUへの割込を発生することを特徴とする集積回路装置。

## 【請求項 3】

請求項 1 において、

前記データ送信ユニットは、前記アクノリッジ検出ユニットが前記アドレスアクノリッジ信号を検出した場合に発生するCPUへの割込発生後に、前記送受信バッファに格納された所定単位のデータのシリアル送信を開始することを特徴とする集積回路装置。

## 【請求項 4】

請求項 1 において、

前記シリアル転送バスは、1本のデータ線と1本のクロック線とを有し、前記データの転送は、前記クロック線にドライブされるクロックに同期して、前記1本のデータ線上にデータ転送がシリアルに行われることを特徴とする集積回路装置。

## 【請求項 5】

シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記送受信バッファに格納されたデータを送信するデータ送信ユニットと、

前記データ送信ユニットが、マスターとして、スレーブデバイスを特定するアドレスをシリアル送信するアドレスフェーズ中に、他のマスターが同時にアドレスを送信することに起因してアービトレーションロストが発生したか否かを検出するアービトレーションロスト検出ユニットとを有し、

前記アービトレーションロスト検出ユニットが、前記アドレスフェーズ中に前記アービトレーションロストの発生を検出しなかった場合に、当該アドレスフェーズ後、前記CPUが、前記送受信バッファに転送対象データを格納することを特徴とする集積回路装置。

## 【請求項 6】

請求項 5 において、

前記アービトレーションロスト検出ユニットが、前記アドレスフェーズ中に前記アービ

トレーションロストの発生を検出した場合に、前記CPUは、前記送受信バッファに前記転送対象データを格納しないことを特徴とする集積回路装置。

【請求項7】

シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し、前記送受信バッファに格納するデータ受信ユニットと、

前記所定単位のデータを受信に応答して、送信側デバイスにデータアクノリッジ信号を送信するアクノリッジ信号生成ユニットとを有し、

前記アクノリッジ信号生成ユニットは、受信可能なデータ単位数に達するまで、前記所定単位のデータを受信するたびに、前記CPUに割込を発生することなく前記データアクノリッジ信号を送信し、前記受信可能なデータ単位数に達した時に、前記データアクノリッジ信号を送信しないことを特徴とする集積回路装置。

【請求項8】

請求項7において、

更に、前記受信可能なデータ単位数を格納する受信数レジスタと、前記データ受信ユニットが前記所定単位のデータを受信するたびに、前記受信数をカウントするカウンタとを有する受信制御ユニットを有し、

前記アクノリッジ信号生成ユニットは、当該カウンタのカウント値が前記受信数レジスタに格納された受信可能なデータ単位数に達することに応答して、前記データアクノリッジ信号を送信しないことを特徴とする集積回路装置。

【請求項9】

シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し前記送受信バッファに格納すると共に、前記送受信バッファに格納されたデータを送信するデータ送受信ユニットと、

前記データ送受信ユニットによる転送データの送受信が行われている間に、前記CPUから前記送受信バッファへのアクセスが行われた場合に、当該アクセス発生フラグを格納するアクセスフラグレジスタとを有することを特徴とする集積回路装置。

## 【書類名】明細書

【発明の名称】シリアル転送バス用の送受信マクロを有する集積回路装置

## 【技術分野】

## 【0001】

本発明は、シリアル転送バス用の送受信マクロを有する集積回路装置に関し、特に、送受信バッファを内蔵して送受信処理中のCPUへの割込頻度を減らしてCPUの負担を小さくすることができる送受信マクロに関する。

## 【背景技術】

## 【0002】

マイクロコンピュータは、CPU、プログラムROM、RAM、外部バスインターフェースなどと共に、リソースとして、タイマー、ADコンバータ、シリアル転送バス用の送受信マクロなどを内蔵する。外部バスインターフェースは、32ビットなど複数ビットの外部バスとのインターフェース機能を有するものであるのに対して、シリアル転送バス用の送受信マクロは、1本のデータ線と1本のクロック線とからなるシリアル転送バスに対するインターフェース機能を有する。

## 【0003】

シリアル転送バスには、例えばフィリップス社が提唱しているI2C規格がある（例えば、特許文献1参照）。このI2C規格のシリアル転送バスは、クロックに同期して、1本のデータ線を経由してアドレスやデータをシリアル転送する。その規格によれば、マスター側の送受信マクロがクロックとデータの2ビットでスタート状態とストップ状態をスレーブ側に送信し、スタート状態後に送信側の送受信マクロがクロックに同期してアドレスやデータをシリアル転送する。そして、1バイトのデータ転送毎に受信側の送受信マクロがアクノリッジ信号をデータ線で返信する。したがって、送信側と受信側の送受信マクロは、データ転送毎にアクノリッジ信号の送出と確認のためにCPUに割込をかける必要がある。

## 【0004】

かかるシリアル転送バスは、電源投入時や所定の処理のセッション開始時において、LSI間でデータを転送して所定のデータ設定を行うなどの用途に利用されるものであり、限られたバスのハードウェアリソースを有効に利用して、データ転送を実現するものとして広く利用されている。

【特許文献1】特開2000-242573号公報

## 【発明の開示】

## 【発明が解決しようとする課題】

## 【0005】

従来のシリアル転送バスは、転送データ量がそれほど大きくない場合は、CPUへの割込頻度はそれほど多く発生せずCPUの負担はそれほど大きくなかった。しかしながら、近年において転送データ量が増大してくると、従来の送受信マクロのように、1バイトのデータ転送毎にCPUに割込をかけて送受信制御を行わせる方法は、CPUへの負担が増大して好ましくない。

## 【0006】

そこで、本発明の目的は、CPUの負担を増大させることなく転送データ量を増やすことができるシリアル転送バス用の送受信マクロを有する集積回路装置を提供することにある。

## 【課題を解決するための手段】

## 【0007】

上記の目的を達成するために、本発明の第1の側面は、シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記所定単位のデータの送信に 응답して受信側デバイスから送信されるデータアクノリッジ信号を検出するアクノリッジ信号検出ユニットと、

前記アクノリッジ信号検出ユニットによる前記データアクノリッジ信号の検出に 응답して、前記CPUへの割込を発生することなく前記送受信バッファに格納されたデータを送信するデータ送信ユニットとを有し、

前記アクノリッジ検出ユニットは、前記所定単位のデータの送信に対応して受信側デバイスから送信されるデータアクノリッジ信号を検出しない場合に、前記CPUにデータアクノリッジ信号未検出の割込を発生することを特徴とする。

#### 【0008】

上記の目的を達成するために、本発明の第2の側面は、シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記送受信バッファに格納されたデータを送信するデータ送信ユニットと、

前記データ送信ユニットが、マスターとして、スレーブデバイスを特定するアドレスをシリアル送信するアドレスフェーズ中に、他のマスターが同時にアドレスを送信することに起因してアービトレーションロストが発生したか否かを検出するアービトレーションロスト検出ユニットとを有し、

前記アービトレーションロスト検出ユニットが、前記アドレスフェーズ中に前記アービトレーションロストの発生を検出しなかった場合に、当該アドレスフェーズ後、前記CPUが、前記送受信バッファに転送対象データを格納することを特徴とする。

#### 【0009】

上記の目的を達成するために、本発明の第3の側面は、シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し、前記送受信バッファに格納するデータ受信ユニットと、

前記所定単位のデータの受信に 응답して、送信側デバイスにデータアクノリッジ信号を送信するアクノリッジ信号生成ユニットとを有し、

前記アクノリッジ信号生成ユニットは、受信可能なデータ単位数に達するまで、前記所定単位のデータを受信するたびに、前記CPUに割込を発生することなく前記データアクノリッジ信号を送信し、前記受信可能なデータ単位数に達した時に、前記データアクノリッジ信号を送信しないことを特徴とする。

#### 【0010】

上記の目的を達成するために、本発明の第4の側面は、シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し前記送受信バッファに格納すると共に、前記送受信バッファに格納されたデータを送信するデータ送受信ユニットと、前記データ送受信ユニットによる転送データの送受信が行われている間に、前記CPUから前記送受信バッファへのアクセスが行われた場合に、当該アクセス発生フラグを格納するアクセスフラグレジスタとを有することを特徴とする。

【発明の効果】

【0011】

上記第1の側面によれば、送信側の時に、送受信マクロは、受信側からのデータアクノリッジを受信しなくなるまで、CPUへの割込を行うことなく、複数単位のデータを連続して送信することができるので、CPUへの負担を軽減して、多くのデータをシリアル転送することが可能になる。

【0012】

上記第2の側面によれば、送信マクロがアドレスフェーズ中にバスアービトレーションをチェックし、アドレスフェーズ中にアービトレーションロストが発生しなかったことを確認してから、送受信バッファに送信用データを格納するので、アービトレーションロストに起因して送受信バッファに格納した送信用データが無駄になることを防止できる。

【0013】

更に、上記第3の側面によれば、受信側の時に、送受信マクロは、予め設定された受信可能なデータ単位数に達するまで、CPUへの割込を行うことなく、複数単位のデータを連続して受信することができるので、CPUへの負担を軽減して、多くのデータをシリアル転送することができる。

【0014】

また、上記第4の側面によれば、データ転送中に誤ってCPU側から送受信バッファにアクセスが発生したことを記憶するアクセスフラグレジスタを設けているので、そのアクセスフラグレジスタを利用することで、ソフトウェアのデバッグ工程を効率的に行うことができる。

【発明を実施するための最良の形態】

【0015】

以下、図面にしたがって本発明の実施の形態について説明する。但し、本発明の技術的範囲はこれらの実施の形態に限定されず、特許請求の範囲に記載された事項とその均等物まで及ぶものである。

【0016】

図1は、本実施の形態におけるシリアル転送バスを示す図である。シリアル転送バス10は、1本のデータ線SDAと1本のクロック線SCLとからなる。つまり、1チャンネルのシリアル転送バスはデータ線とクロック線からなる1対のバスで構成される。したがって、複数チャンネルを有する場合は、上記のデータ線SDAとクロック線SCLからなる1対のバスが、複数対設けられる。図1には、マイクロコンピュータAとBと、メモリ12と、所定の機能を有する機能デバイス14とが、バス10を介して接続されている。このように、本実施の形態におけるシリアル転送バスは、複数のデバイス（集積回路装置）を接続してクロックに同期して1本のデータ線でデータをシリアルに転送する。

【0017】

シリアル転送バス10を介してデータの転送を行う送受信マクロが、マイクロコンピュータA、Bや、メモリ12、機能デバイス14に内蔵されていて、この送受信マクロがシリアル転送バス10とのインターフェース機能を有する。

【0018】

マイクロコンピュータAがメモリ12など他のデバイスに対してデータの書き込みを行う場合は、マイクロコンピュータAがマスター側になり、スレーブ側になるメモリ12を特定するアドレスをバス上に送出する。それに応答してメモリ12からアクノリッジが返信されると、マイクロコンピュータAは送信側になり、受信側のメモリ12にデータを転送する。第2に、マイクロコンピュータAがマイクロコンピュータBや機能デバイス14



など他のデバイスのデータを読み出す場合は、マイクロコンピュータ A がマスター側になり、スレーブ側デバイスを特定するアドレスを転送する。それに応答してスレーブ側デバイスからアクノリッジが返信されると、マイクロコンピュータ A は受信側になり、送信側のデバイスが送信するデータを受信する。

#### 【0019】

上記のデータ転送は、例えば、システムの電源投入時や、所定のセッションやフェーズ毎に実行され、マイクロコンピュータ A が他のデバイスにデータを送信したり、他のデバイスからデータを受信したりする。

#### 【0020】

図 2 は、本実施の形態における集積回路装置であるマイクロコンピュータの全体構成図である。マイクロコンピュータは、所定の処理を実行する CPU と、CPU が実行するプログラムが格納された ROM と、CPU が処理実行中にデータを一時的に格納する RAM と、外部のバス 23 とのインターフェースを行う外部バスコントローラ 22 と、外部のメモリとのバス 25 のインターフェースを行うメモリインターフェース 24 とが、高速の CPU バス 20 を介して接続される。また、マイクロコンピュータは、周辺リソース 30 として、シリアル転送バス 10 とのインターフェースを行う送受信マクロ 32 と、AD コンバータ 34 と、タイマー 36 などとを有し、これらの周辺リソース 30 は、低速バス 28 を介して接続されている。また、低速バス 28 は、バスコンバータ 26 を介して高速の CPU バス 20 に接続される。CPU バス 20 は例えば 32 ビットで構成され、低速バス 28 はそれより狭いバス幅、例えば 16 ビットで構成される。

#### 【0021】

また、高速バス 20 に接続される外部バスコントローラ 22 やメモリインターフェース 24 は、高速クロックに同期して動作するのに対して、低速バス 28 に接続される送受信マクロ 32 は、低速クロックに同期して動作する。

#### 【0022】

CPU は、ROM 内のプログラムを実行して、所定の処理を実行する。また、CPU は、シリアル転送バス 10 を介してデータ転送を行う場合は、送受信マクロ 32 を制御してデータ転送を実行させる。本実施の形態では、転送データ量を大きくするために、送受信マクロ 32 内に例えば FIFO からなる送受信バッファを設け、CPU がこの送受信バッファにデータを格納してデータ転送を指令すると、送受信マクロ 32 が、CPU への割込を頻繁に発生することなくデータのシリアル転送を実行する。但し、送受信マクロ 32 は、シリアル転送バス 10 の特有の性質に対応すべく、CPU への割込発生頻度を抑えるための種々の改良構成を有する。

#### 【0023】

##### 〔第 1 の実施の形態〕

図 3 は、本実施の形態における送受信マクロの構成図である。送受信マクロ 32 は、データ転送量を増大させるために、複数の送信データを格納したり複数の受信データを格納する送受信バッファとして FIFO 46 を有し、低速バス 28 を介して CPU からアクセス可能に構成される。また、アドレスやデータの送受信手段として、データ送信・受信ユニット 48 を有し、CPU はレジスタ REG4 を介して、このユニット 48 にデータ送信又は受信を指令する。データ送信・受信ユニット 48 は、データ送信の時は、FIFO 46 内のパラレルデータをパラレル・シリアル変換して、データをバス 10 にシリアルに送信し、データ受信の時は、バス 10 からシリアルに受信したシリアルデータをシリアル・パラレル変換して、受信データを FIFO 46 内に格納する。データのシリアル転送は、クロック線 SCL に出力されるクロックに同期して 1 ビットずつ行われる。データ送信・受信ユニット 48 は、低速バス 28 を経由して CPU によりアドレスやデータを直接設定される場合もある。

#### 【0024】

送受信マクロ 32 は、シリアル転送バス 10 のクロック線 SCL に転送用の同期クロックを送信したり、受信したりするクロック送信・受信ユニット 40 を有し、CPU から指定されレジスタ REG1 に格納されるクロック速度に対応するクロックを生成して送信し、また

クロックを受信する。スタート・ストップ状態発生ユニット42は、データSDAとクロックSCLの2つの信号により、データ転送のスタート状態とストップ状態とを発生し、送信する。マイクロコンピュータ32がマスターモードの時に、スタート・ストップ状態発生ユニット42が、バス10をデータ転送のスタート状態にドライブし、データ送信・受信ユニット48が、スレーブデバイスを特定するアドレスをシリアル送信する。その後、マスター側とスレーブ側とでデータ転送（スレーブデバイスへのリードまたはライト）が行われ、データ転送を終了する場合は、スタート・ストップ発生ユニット42は、データ転送のストップ状態を出力する。別のデータ転送を開始する場合は、再度スタート状態を出力する。スタート・ストップ状態発生ユニット42は、CPUからレジスタREG2に設定される指令に応答して、それぞれの状態にバス10をドライブする。

#### 【0025】

スタート・ストップ状態検出ユニット44は、シリアル転送バス10に出力されたスタート状態またはストップ状態を検出する。そして、スタート状態の後に転送されるアドレスが自分のアドレスに対応するかがチェックされる。スタート・ストップ状態検出ユニット44は、検出した状態データを、レジスタREG3に格納し、CPUにその状態を通知する。

#### 【0026】

スタート・ストップ状態発生ユニット42及び検出ユニット44によれば、シリアル転送バス10にデータ又はアドレス転送中か否かを監視することができる。そこで、転送中フラグレジスタ58が設けられ、データ転送中であれば、データ送信・受信ユニット48からのデータ転送開始信号S482に基づき、転送中フラグがこの転送中フラグレジスタ58に書き込まれる。

#### 【0027】

アービトレーションロスト検出ユニット50は、マスターモード時に、データ送信・受信ユニット48からアドレスがシリアルに出力されるアドレスフェーズ中に、他のデバイスからそのアドレスと競合するアドレスが同時に出力されてバス権取得の仲裁手続でバス権を失ったか否かを検出する。この仲裁手続については、後に詳述する。検出したアービトレーションロストの発生は、レジスタREG5を介してCPUに通知される。

#### 【0028】

アクノリッジ発生ユニットであるACK発生ユニット52は、データフェーズで、例えば1バイトのデータを受信した後に無事に受信したことを送信側デバイスに知らせるために、アクノリッジ信号をデータ線SDAに出力する。このアクノリッジ信号の出力は、後述するとおりCPUへの割込をすることなく行われる。或いは、ACK発生ユニット52は、アドレスフェーズで、受信したアドレスが自分のアドレスに該当する場合も、マスター側デバイスにアドレスの一致を知らせるために、アクノリッジ信号をデータ線SDAに出力する。このアクノリッジ信号の出力指令は、レジスタREG6を介してCPUから行われる。

#### 【0029】

アクノリッジ検出ユニットであるACK検出ユニット54は、アドレスフェーズでスレーブ側デバイスから送信されるアクノリッジ信号を検出したり、データフェーズで受信側デバイスから送信されるアクノリッジ信号を検出する。ACK検出ユニット54は、データフェーズでアクノリッジ信号を受信しなかったことを検出して、CPUへの未検出の割込IRQ信号を発生し、クロックイネーブル信号CLKENにより、クロック送信・受信ユニット40にクロック線SCLをLレベルにドライブさせてバスをウェイト状態にし、次のデータ転送が行われないようにする。

#### 【0030】

また、受信制御ユニット56は、後述するとおり、受信データ数をCPUから設定され、その設定された受信データ数だけ連続してデータを受信した時に、CPUに割込を発生し、アクノリッジ信号送信許可フラグ（図示せず）の状態に応じて、ACK発生ユニット52がアクノリッジの発生を禁止（または許可）する。

#### 【0031】

図 4 は、シリアル転送バスのドライブ回路の一例を示す図である。図 4 では、図 1 に示した 4 つのデバイスのドライブ回路が示されている。各デバイスのドライブ回路は、シリアル転送バスのデータ線 SDA（又はクロック線 SCL）にドレイン端子が接続され、ソースが接地されたオープンドレイントランジスタ QA、QB、Q12、Q14 からなる。そして、データ線 SDA（又はクロック線 SCL）は、プルアップ抵抗 R を介して電源 Vdd に接続されている。したがって、全てのデバイスのドライブトランジスタが駆動しない場合は、バスのデータ線 SDA（又はクロック線 SCL）は H レベルに維持され、いずれかのデバイスのドライブトランジスタが駆動される場合は、バスのデータ線 SDA（又はクロック線 SCL）は L レベルにされる。かかるドライブ回路は、クロック送信・受信ユニット 4 0、スタート・ストップ状態発生ユニット 4 2、データ送信・受信ユニット 4 8、ACK 発生ユニット 5 2 内に設けられる。

#### 【0032】

また、各デバイスの送受信マクロには、バスのデータ線 SDA（又はクロック線 SCL）の電圧レベルを検出する回路が設けられ、バスの電圧レベルの H、L レベルが検出される。そのような検出回路は、クロック送信・受信ユニット 4 0、スタート・ストップ状態検出ユニット 4 4、データ送信・受信ユニット 4 8、アービトレーションロスト検出ユニット 5 0、ACK 検出ユニット 5 4 などに設けられている。

#### 【0033】

図 5 は、シリアル転送バスの信号波形図である。図 5 は、I2C 規格のバス仕様書から転用したものである。図 5 には、データ線 SDA の信号波形とクロック線 SCL の信号波形とが示されている。マスターモードの時に、送受信マクロのスタート・ストップ状態発生ユニット 4 2 が、クロック線 SCL を H レベルに維持しながらデータ線 SDA を H レベルから L レベルに駆動することで、スタート状態がバス上に通知される。それに続いて、マスター側の送受信マクロが、クロック線 SCL を L レベルに駆動した後にデータ線 SDA を H 又は L レベルにし、クロック線 SCL を L レベルから H レベルにするときにそのデータ線 SDA の状態を維持して、スレーブ側にアドレス信号を認識させる。同様に、クロック SCL に同期して 1 バイトのアドレスがシリアルに出力される。この 1 バイトのアドレスの中に、リードかライトかの動作コマンドが含まれる。これがアドレス転送のアドレスフェーズである。

#### 【0034】

1 バイトのアドレスが出力されると、アドレスで特定されたスレーブ側の送受信マクロがアクノリッジ信号として 9 番目のクロックに同期してデータ線 SDA を L レベルに駆動する。図 4 に示したオープンドレインの駆動回路により、アドレスで特定されていない他のデバイスは、データ線 SDA を H レベルに維持することで、アドレスで特定されたスレーブ側デバイスによる L レベルのアクノリッジ信号が正しくマスタ側に伝えられる。

#### 【0035】

このスレーブ側によるアクノリッジ信号によりアドレスフェーズは終了する。そして、データフェーズが開始され、マスター側からのクロック信号 SCL に同期して、送信側デバイスから 1 バイトのデータがシリアルに出力されて、データ転送が行われる。動作コマンドがリードの場合は、スレーブ側が送信側になり、ライトの場合はマスター側が送信側になる。そして、1 バイト（8 ビット）のデータがシリアル転送されると、受信側はデータ線 SDA に L レベルのアクノリッジ信号を送信する。受信側が L レベルのアクノリッジ信号を送信すると、送信側の送受信マクロは、それに応答して必要なデータのシリアル転送を継続する。受信側がデータ線 SDA を L レベルに駆動しなければ、アクノリッジ信号が出されなかったことになり、それにより送信側デバイスに、データ転送を終了すべきことが通知される。その後、マスタ側デバイスがデータフェーズを終了すべきか否かを判断する。

#### 【0036】

マスター側の送受信マクロは、全てのデータ転送が終了すると、クロック線 SCL を H レベルに維持しながらデータ線 SDA を L レベルから H レベルに駆動してストップ状態を出力する。ストップ状態を出力するとマスターモードが終了する。あるいは、マスター側の送受信マクロは、全てのデータ転送が終了すると、クロック線 SCL を H レベルに維持しながら

らデータ線SDAをHレベルからLレベルに駆動して再度スタート状態を出力することでもできる。その場合は、マスター側デバイスは、マスターモードを維持しながら、別のデータ転送を開始する。

#### 【0037】

このように、1バイトのアドレス転送後にスレーブ側がアクノリッジ信号を送信し、1バイトのデータ転送後には受信側がアクノリッジ信号を送信する。マスター側または送信側は、これらのアクノリッジ信号を検出して、次のデータ転送を行う。また、マスターモードは、スタート状態の出力からストップ状態の出力まで維持される。尚、バス権取得のための仲裁手続は、アドレス転送中に行われるが、その動作については後述する。

#### 【0038】

図6は、第1の実施の形態を説明するタイミングチャート図である。本実施の形態では、送受信マクロ内にFIFOからなる送受信バッファを設け、CPUが複数バイトのデータをこの送受信バッファに格納することで、データ転送量を増大させている。しかしながら、シリアル転送バスは、規格上、1バイトのデータをシリアル転送するたびに受信側がデータアクノリッジ信号を送信し受信側がそれを確認し、更に次の1バイトのデータをシリアル転送する。そして、受信側がデータアクノリッジ信号を送信しない場合は受信側はそれ以上のデータ転送を行わない。このように、複数バイトのデータを転送する場合1バイト転送毎にデータアクノリッジ信号の送信と確認を行う必要があり、そのたびにCPUに割り込みをかけると、CPUの負担が増大し、FIFOを設けてデータ転送量を増大させようとする目的に沿わなくなる。

#### 【0039】

そこで、本実施の形態では、アドレスフェーズでは1バイトのアドレスをシリアル転送しそれに対するアクノリッジ信号を確認し、その後のデータフェーズでは、送受信マクロが受信側からのデータアクノリッジ信号を確認すれば、CPUへの割り込みを発生することなく、次の1バイトのデータのシリアル転送を行い、データアクノリッジ信号を確認できない場合にのみCPUへの割り込みを発生し、データフェーズを終了するようにする。

#### 【0040】

図6に沿って具体的な動作を説明する。このタイミングチャートでは、マスター側が送信側に、スレーブ側が受信側になり、送信側デバイス100と受信側デバイス200との間でデータ転送を行う例であり、送信側デバイス100の送受信マクロ132と受信側デバイス200の送受信マクロ232との間でデータ転送が行われる。

#### 【0041】

最初に、送信側デバイス100がマスターモードになり(S10)、CPUからの指令にตอบสนองして、送信側送受信マクロ132のスタート・ストップ状態発生ユニット42が、バスをスタート状態にドライブする(S11)。それに続いて、送信側送受信マクロ132が、1バイトのアドレスを送信し(S12)、そのアドレスで特定されたスレーブ側のデバイスの送受信マクロ232がアドレスアクノリッジ信号を出力する(S13)。送受信マクロ132は、このアドレスアクノリッジ信号を検出すると、CPUに割り込みをかけ(S14)、それにตอบสนองして、CPUはデータ転送を指令する(S15)。

#### 【0042】

これにより送信側送受信マクロ132は、データフェーズに入り、FIFO内に格納されている1バイトのデータをシリアル転送する(S16)。1バイトのデータシリアル転送が行われると、受信側送受信マクロ232は、データアクノリッジ信号を送信する(S17)。送信側の送受信マクロ132内のACK検出ユニット54は、このデータアクノリッジ信号を検出すると、CPUへの割り込みをかけることなく、アクノリッジ検出信号S541によって、データ送信・受信ユニット48に次のデータ転送を指示する。これにตอบสนองして、データ送信・受信ユニット48は、次の1バイトのデータをFIFOから読み出しパラレル・シリアル変換して、バスにシリアル転送を行う(S18)。1バイトのデータ転送が行われると、受信側送信マクロ232は、再度データアクノリッジ信号を送信する(S19)。前述と同様に、送信側の送受信マクロのACK検出ユニット54は、このデータ

アクノリッジ信号を検出すると、CPUに割り込みをかけることなく、データ送信・受信ユニット48に次のデータ転送を指示する。これに回答して、次の1バイトのデータがFIFOから読み出されシリアル転送される(S20)。

#### 【0043】

図6の例では、3バイトのデータ転送後において、受信側の送受信マクロはデータアクノリッジ信号を送信しないで、データ転送を終了すべきことを通知する(S21)。送信側のマクロのACK検出ユニット54は、データアクノリッジ信号が送信されなかったことを検出し、CPUへのデータアクノリッジ信号未確認の割り込み信号IRQを発生すると共に、クロック送信・受信ユニット40へのクロックイネーブル信号CLKENをディセーブル状態にして、クロック送信・受信ユニット40によりクロック信号線SCLをLレベルにドライブさせ、バスをウェイト状態にさせる。それにより、それ以降のクロック信号の送出が停止し、次のデータ転送が行われなくなる(S22)。CPUは、データアクノリッジ信号未確認の割り込み信号IRQに回答して、データフェーズの終了を認識し、データ転送を終了する場合は、送受信マクロ132にストップ状態をドライブすることを指令する。送受信マクロ132のスタート・ストップ状態発生ユニット42は、これに回答して、バスをストップ状態にドライブする(S23)。以上で、データ転送が完了する。あるいは、CPUは、割込信号に回答して、再度スタート状態の発生を送受信マクロのスタート・ストップ状態発生ユニット42に指令して、別のデータ転送を開始することもある。その場合は、図中の工程S11以下の動作が繰り返される。

#### 【0044】

以上のように、本実施の形態では、データフェーズにおいてFIFO内の複数バイトのデータを転送する場合、送受信マクロは、1バイト転送後のデータアクノリッジ信号を検出した時はCPUへの割り込みを発生することなく自動的に次の1バイト転送を実行する。そして、送受信マクロは、データアクノリッジ信号を検出しない時に初めてCPUに割り込みを発生して、データフェーズの終了を通知する。従って、複数バイトのデータを連続して転送する場合でも、CPUへの割り込み頻度が減り、CPUへの負担を軽減することができる。

#### 【0045】

##### [第2の実施の形態]

図7は、バスアービトレーション手続きを説明するための図である。この図は、I2C規格のバス仕様書から転用したものである。図7(A)に示されるように、本実施の形態におけるシリアル転送バスは複数のマスターデバイスとスレーブデバイスを接続する。そして、任意のタイミングでいずれかのマスターデバイスがスタート状態にドライブしてアドレスを送信することを許している。従って、通常は最初にスタート状態にドライブしたマスターデバイスがバス権を取得することになるが、まれに、複数のマスターデバイスが同時にスタート状態にドライブする場合がある。

#### 【0046】

図7(B)の波形図では、マスターAとBが同時にバスをスタート状態にドライブし、同じタイミングでアドレスを送信開始した例である。図4で説明したとおり、ドライブ回路はオープンドレイントランジスタであるので、いずれかのデバイスがバスをLレベルにドライブすると、他のデバイスのドライブ状態にかかわらずバスはLレベルにドライブされる。

#### 【0047】

図7(B)の例では、クロックCL1、CL2では、マスターA、Bが同じアドレス信号H、Lをドライブしているが、クロックCL3では、マスターAがHレベル、マスターBがLレベルをドライブしたため、両者のアドレス間でコンフリクトが発生している。これに伴い、マスターAの送受信マクロ内のアービトレーションロスト検出ユニット50がバスアービトレーション手続きでバス権を失ったことを検出し、レジスタREG5を介してCPUにアービトレーションロストを通知する。これに回答して、CPUはその後のデータフェーズでのデータ転送の実行を中止する。

## 【0048】

図8は、本実施の形態における送受信マクロの動作フローチャート図である。図中、右側が本実施の形態の動作を示し、左側はそれに対する比較例の動作を示す。本実施の形態では、CPUがマスターモードになり（S30）、送受信マクロがアドレスの送信を開始する（S32）。送受信マクロがアドレスフェーズ中に前述のアービトラクションロストを検出しなければ（S34）、有効にバス権を取得したものとみなされ、その後、CPUはFIFOに送信すべきデータを書き込む（S36）。そして、データ書き込み後に、マスターモードで複数バイトのデータ転送を行う（S38）。また、アドレスフェーズ中にアービトラクションロストを検出すると（S34）、送受信マクロはスレーブモードになり（S40）、いずれかのマスターデバイスからアドレスを受信すると（S42）、スレーブモードでデータ転送を実行する（S44）。その後、適宜マスターモードに切り換えられ（S30）、必要なデータ転送の手順を再度実行する。

## 【0049】

上記の本実施の形態の動作に対して、比較例では、マスターモードになると（S30）、その時点でCPUが送受信マクロのFIFOに送信データを書き込む（S31）。その後、送受信マクロはアドレスを送信し（S32）、アービトラクションロストの発生をチェックする。もし、アービトラクションロストが検出されると（S34）、送受信マクロはスレーブモードに待機し（S40）、FIFOに書き込まれた送信データをクリアする（S41）。そして、アドレスを受信すると（S42）、スレーブモードでデータ転送を実行する。アービトラクションロストを検出しない場合は、上記と同じである。

## 【0050】

このように、比較例では、アドレスフェーズでのアービトラクションロストの有無を確認するまえに、FIFOに転送データを書き込んでいるので、アービトラクションロストが検出されると、書き込んでいたFIFO内のデータが無駄になってしまう。それに対して、本実施の形態では、シリアル転送バスではアドレスのシリアル転送中にバス権獲得のアービトラクション手続きを行うので、そのバス権を獲得したことを確認した後に、FIFOへの転送データの書き込みを実行している。従って、FIFO内への転送データの書き込みが無駄になることはない。

## 【0051】

図9は、本実施の形態におけるシーケンスチャート図である。図6と同じ工程には同じ工程番号が与えられている。この例は、アドレスフェーズでアービトラクションロストが発生しなかった例である。送信側のデバイス100がマスターモードになり（S10）、送受信マクロ132がバスをスタート状態にドライブしてから（S11）、アドレスフェーズでスレーブのアドレスをシリアル送信する（S12）。このアドレスフェーズの間にアービトラクションロストが発生せずに、正常にスレーブデバイスからアクノリッジ信号を受信すると（S13）。送受信マクロ132は、CPUに割り込みIRQをかけ（S14）、それに応答して、CPUは初めてマクロ内のFIFOに転送データを書き込む（S15A）。その後、CPUによるデータ転送指令に応答して（S15）、送受信マクロ132は、FIFO内の1バイトのデータをシリアル転送する（S16）。その後は、図6と同じ動作であり、1バイト転送が完了するたびに、受信側からデータアクノリッジ信号を受信し、CPUに割り込みをかけることなく、次の1バイトデータをシリアル転送する。一連のデータ転送は、受信側からデータアクノリッジ信号を受信しなくなるまで継続される。

## 【0052】

図10は、本実施の形態における別のシーケンスチャート図である。この図においても、図6、図9と同じ工程には同じ工程番号を与えている。この例は、アドレスフェーズ中にアービトラクションロストが発生した例である。送信側デバイス100がマスターモードになった後（S10）、CPUからの指令に応答して送受信マクロ132は、バスをスタート状態にドライブし（S11）、1バイトのアドレスをシリアル転送する（S12）。このアドレスフェーズ中にアービトラクションロストが発生したことを、送受信マクロ132内のアービトラクションロスト検出ユニット50が検出すると（S50）、それが

CPUに通知され(S51)、CPUは一旦スレーブモードになり待機する(S52)。所定時間経過してもスレーブとしてのアドレスを受信しない場合に、再度、CPUはマスターモードになり(S53)、送受信マクロ132に再度アドレスフェーズでアドレス転送することを指令する。これに应答して、送受信マクロ132は、バスをスタート状態にドライブし(S54)、引き続いて1バイトのアドレスをシリアル転送する(S55)。今回はこのアドレスフェーズ中にアービトレーションロストが発生せず、したがって、CPUはFIFOに転送データを書き込む(S15A)。その後の動作は、図9と同じである。

#### 【0053】

このように、本実施の形態では、アドレスフェーズ中にアービトレーションロストが発生しても、FIFO内に転送データは書き込まれていないので、FIFO内のデータが無駄にクリアされることはない。

#### 【0054】

##### 〔第三の実施の形態〕

送受信マクロ内に複数バイトのデータを格納できる送受信バッファとしてFIFOを設けているので、複数バイトのデータを連続して転送することができる。上記では、送信側のマクロがCPUへの割込を発生することなく連続してデータを転送することを説明した。第3の実施の形態は、受信側のマクロがCPUへの割込を発生することなく連続してデータを受信できる構成に関する。

#### 【0055】

図11は、本実施の形態における送受信マクロの一部の構成を示す図である。送受信バッファ46は、FIFOとそれを制御するためのライトポインタ461とリードポインタ462とを有する。ライトポインタは、FIFO内の新たに書き込まれるべきアドレスを格納し、リードポインタは、FIFO内の新たに読み出されるべきアドレスを格納する。受信制御ユニット56は、受信レジスタ561と、ダウンカウンタ562と、ゼロ検出回路563とを有する。

#### 【0056】

CPUは、データ受信開始前に受信可能なバイト数を受信レジスタ561に設定すると共に、アクノリッジ発生ユニット54のレジスタREG6にアクノリッジ信号発生許可・禁止フラグを許可状態に設定する。その設定された受信バイト数はダウンカウンタ562に格納され、データ送信・受信ユニット48が、1バイトのデータを正常に受信するたびに、受信完了信号S481を出力し、ダウンカウンタ562は設定値をダウンカウントする。また、ACK発生ユニット54は、受信完了信号S481に应答して、データアクノリッジ信号を発生し、転送バス10に出力する。そして、受信可能なバイト数のデータを受信完了した時点で、ダウンカウンタ562のカウント値はゼロになり、ゼロ検出回路563がそれを検出し、CPUに割込IRQを発生すると共に、カウントゼロ信号S561をACK発生ユニット54に出力する。このカウントゼロ信号S561に应答して、ACK発生ユニット54は、データアクノリッジ信号発生許可・禁止フラグが許可状態であることを確認して、データアクノリッジ信号を発生せず、送信側デバイスにはデータアクノリッジ信号は送信されない。したがって、送信側デバイスは、それ以後のデータ転送を行わない。

#### 【0057】

このように、CPUは、受信セッションを開始するたびに、受信レジスタ561に連続受信バイト数を設定すると共に、連続受信後にアクノリッジ信号を送信するかしないかをデータアクノリッジ信号発生許可・禁止フラグを設定して、受信動作を制御する。

#### 【0058】

図12は、本実施の形態におけるシーケンスチャート図である。図9と同じ工程には同じ引用番号を与えている。この例は、受信側デバイス200が3バイトのデータ受信を行ってデータ転送を終了する例である。

#### 【0059】

送信側デバイス100のCPUがマスターモードになり(S10)、送受信マクロ132がバスにスタート状態をドライブし(S11)、アドレスをシリアル転送すると(S1



2)、受信側の送受信マクロ232が受信したアドレスが自分のアドレスであることを認識して、CPUに割込をかける(S13A)。CPUは、受信数レジスタに受信可能なバイト数を設定し(S13B)、送受信マクロ232は、アドレスアクノリッジ信号を出力する(S13)。

#### 【0060】

送信側デバイスの送受信マクロ132は、アドレスアクノリッジ信号を検出して、CPUに割込をかけ(S14)、CPUは、それに応答してFIFOに転送データを書き込むとともに(S15A)、データ送信・受信ユニット48にデータ転送の開始を指示する(S15)。1バイトのデータが転送されると(S16)、受信側の送受信マクロ232のデータ送信・受信ユニットが受信完了信号S481を出力し、ACK発生ユニットがデータアクノリッジ信号を出力すると共に(S17)、ダウンカウンタで設定値のダウンカウントがおこなわれる(S17A)。同様のデータ転送とデータアクノリッジ信号の返信及びダウンカウントが行われ(S18, S19, S19A)、最後のデータ転送が行われると(S20)、受信側の送受信マクロ232は、データアクノリッジ信号を返信せず(S21)、それを検出した送信側の送受信マクロ132は、CPUに割込をかける(S22)。そして、ダウンカウントが行われると(S21A)、受信側の送受信マクロ232内の受信制御ユニット56がCPUに割込をかけて(S21B)、CPUは最初に設定したバイト数のデータ受信を完了したことを通知される。そして、最後に、CPUからの指令に応答して(S23)、送受信マクロ132は、バスにストップ状態をドライブする(S24)。

#### 【0061】

このように、受信側になったときも、送受信マクロは、CPUに割込をかけることなく1バイトのデータ受信後にデータアクノリッジ信号を返信することで、CPUへの負担をかけずにデータ転送量を増やすことができる。また、CPUが最初に受信可能なバイト数を設定し、その設定値のデータ受信が完了した時点で、送受信マクロがCPUに受信完了割込をかけ、データアクノリッジ信号を返信しないようにしているので、受信側デバイスが受信データでオーバーフローすることが回避され、CPUは一度に受信すべきバイト数を制御することができる。つまり、CPUは、従来のように、1バイト受信毎に割込をかけられて所定の判断を基準にしてデータアクノリッジ信号送信許可・禁止フラグの設定を行う必要はなく、複数バイトの受信を割込なしで連続して実行させ、それが終了した時点で、データアクノリッジ信号を送信するか否かを制御することができるので、CPUの制御の頻度を下げることができる。また、そのために必要な送受信マクロ内のハードウェア構成は、図11の受信制御ユニット56を設けるだけでよい。

#### 【0062】

##### 【第4の実施の形態】

本実施の形態では、送受信マクロ内にFIFOからなる送受信バッファを設けてCPUへの割込を行わずに複数バイトのデータ転送を行うようにしている。それに伴って、CPUによる送受信バッファへのアクセスを、データ転送が行われるデータフェーズ期間中においては、禁止する仕様にする。データフェーズ期間中のCPUからのFIFOへのアクセスを禁止することで、送受信マクロ内のデータ送信・受信ユニット48によるFIFOへのアクセス構成を簡素化することができる。例えば、FIFOをシングルポートRAMにより構成することが可能になる。

#### 【0063】

上記のCPUによるFIFOへのアクセス禁止は、CPUが実行するソフトウェアにより遵守される。しかしながら、ソフトウェア開発において何らかのバグが含まれる場合は、かかるアクセス禁止の仕様を破って、CPUによるFIFOへの不適切なアクセスが発生する場合がある。このような不適切なアクセスが発生すると、データフェーズ期間中のデータ転送にエラーが発生したり、システムが予測できない状態に陥ることが予想される。

#### 【0064】

そこで、第4の実施の形態では、不適切なアクセスの発生を検出できる機能を送受信マ



クロに設ける。図3に示すとおり、送受信マクロ32は、データフェーズ期間であることを示すデータ転送中フラグ58と、不適切なアクセスの発生を記録するエラーフラグ60とを有する。

#### 【0065】

図9の動作シーケンスチャートを参照しながら、上記フラグ58、60についての動作を説明する。転送中フラグ58には、送信時であれば、CPUがFIFOへの送信データ書き込み終了時に転送中フラグが書き込まれ、FIFO内のエンプティフラグがエンプティになったことに応答して転送完了フラグが書き込まれる。また、転送中フラグ58には、受信時であれば、CPUが受信数カウンタに受信数を書き込んだ時に転送中フラグが書き込まれ、ダウンカウンタのカウント値がゼロになって受信が終了した時に転送完了フラグが書き込まれる。また、転送中フラグ58には、データ送信・受信ユニット48から、データ送信・受信開始信号S482が供給される。したがって、転送中フラグ58には、図9の工程S16からS24までのデータ転送期間中であることを示すフラグが格納される。データ転送期間中には、送信側デバイスとしてデータを送信中の期間と、受信側デバイスとしてデータを受信中の期間とが含まれる。そして、転送中フラグ58が転送完了状態の時、つまりCPUへの割込状態の時に、CPUからFIFOへのアクセスが許可される。

#### 【0066】

送受信バッファであるFIFO46は、CPUからアクセスされるときに、低速バス28を介して供給される送受信バッファのアドレスを検出する。その為に、デコードDECを内蔵する。そして、供給されたアドレスが自分のアドレスであることを認識して、その後のCPUからのアクセスを受ける。そこで、FIFO46は、転送中フラグ58がデータ転送期間中であることを示すフラグを格納し、且つ、CPUからのアクセスを検出すると、エラーフラグ60に不適切なアクセスが発生したことを示すフラグを格納する。このエラーフラグ60は、例えば、デバッグ用プログラムを実行することにより、CPUから読み出される。

#### 【0067】

以上のように、本実施の形態によれば、データ転送期間中にCPUからFIFOへの不適切なアクセスが発生した場合、そのアクセスの発生がエラーフラグ60に記録される。したがって、プログラム開発工程におけるデバッグ工程で、このエラーフラグ60の内容をチェックすることで、プログラムによるFIFOへの不適切なアクセス命令が出されていることを検出することができ、デバッグ工程に大きな寄与をもたらすことができる。

#### 【0068】

以上の実施の形態をまとめると次の付記の通りである。

#### 【0069】

(付記1) シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記所定単位のデータの送信に応答して受信側デバイスから送信されるデータアクノリッジ信号を検出するアクノリッジ信号検出ユニットと、

前記アクノリッジ信号検出ユニットによる前記データアクノリッジ信号の検出に応答して、前記CPUへの割込を発生することなく前記送受信バッファに格納されたデータを送信するデータ送信ユニットとを有し、

前記アクノリッジ検出ユニットは、前記所定単位のデータの送信に対応して受信側デバイスから送信されるデータアクノリッジ信号を検出しない場合に、前記CPUにデータアクノリッジ信号未検出の割込を発生することを特徴とする集積回路装置。

#### 【0070】

(付記2) 付記1において、

前記CPUへのデータアクリッジ信号未検出の割込発生に応答して、前記データ送信ユニットからのデータ送信が終了されることを特徴とする集積回路装置。

【0071】

(付記3) 付記1において、

前記アクリッジ検出ユニットは、

前記データ送信ユニットが、マスターとして、スレーブデバイスを特定するアドレスを送信したことに応答して、前記スレーブデバイスから送信されるアドレスアクリッジ信号を検出した場合に、前記CPUへの割込を発生することを特徴とする集積回路装置。

【0072】

(付記4) 付記1において、

前記データ送信ユニットは、前記アクリッジ検出ユニットが前記アドレスアクリッジ信号を検出した場合に発生するCPUへの割込発生後に、前記送受信バッファに格納された所定単位のデータのシリアル送信を開始することを特徴とする集積回路装置。

【0073】

(付記5) 付記1において、

前記シリアル転送バスは、1本のデータ線と1本のクロック線とを有し、前記データの転送は、前記クロック線にドライブされるクロックに同期して、前記1本のデータ線上にデータ転送がシリアルに行われることを特徴とする集積回路装置。

【0074】

(付記6) シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記送受信バッファに格納されたデータを送信するデータ送信ユニットと、

前記データ送信ユニットが、マスターとして、スレーブデバイスを特定するアドレスをシリアル送信するアドレスフェーズ中に、他のマスターが同時にアドレスを送信することに起因してアービトレーションロストが発生したか否かを検出するアービトレーションロスト検出ユニットとを有し、

前記アービトレーションロスト検出ユニットが、前記アドレスフェーズ中に前記アービトレーションロストの発生を検出しなかった場合に、当該アドレスフェーズ後、前記CPUが、前記送受信バッファに転送対象データを格納することを特徴とする集積回路装置。

【0075】

(付記7) 付記6において、

前記アービトレーションロスト検出ユニットが、前記アドレスフェーズ中に前記アービトレーションロストの発生を検出した場合に、前記CPUは、前記送受信バッファに前記転送対象データを格納しないことを特徴とする集積回路装置。

【0076】

(付記8) シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し、前記送受信バッファに格納するデータ受信ユニットと、

前記所定単位のデータの受信に応答して、送信側デバイスにデータアクリッジ信号を送信するアクリッジ信号生成ユニットとを有し、

前記アクリッジ信号生成ユニットは、受信可能なデータ単位数に達するまで、前記所

定単位 of データを受信するたびに、前記CPUに割込を発生することなく前記データアクノリッジ信号を送信し、前記受信可能なデータ単位数に達した時に、前記データアクノリッジ信号を送信しないことを特徴とする集積回路装置。

【0077】

(付記9) 付記8において、

更に、前記受信可能なデータ単位数を格納する受信数レジスタと、前記データ受信ユニットが前記所定単位 of データを受信するたびに、前記受信数をカウントするカウンタとを有する受信制御ユニットを有し、

前記アクノリッジ信号生成ユニットは、当該カウンタのカウント値が前記受信数レジスタに格納された受信可能なデータ単位数に達することに応答して、前記データアクノリッジ信号を送信しないことを特徴とする集積回路装置。

【0078】

(付記10) 付記9において、

前記CPUは、前記受信数レジスタに前記受信可能なデータ単位数を設定することを特徴とする集積回路装置。

【0079】

(付記11) 付記9において、

前記受信制御ユニットは、前記受信可能なデータ単位数に達することに応答して、前記CPUに割込を発生することを特徴とする集積回路装置。

【0080】

(付記12) シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロを有する集積回路装置において、

所定の処理を実行するCPUを有し、

前記送受信マクロは、

前記CPUからアクセスされ、前記シリアル転送バスに対して送信または受信される複数単位 of データを格納する送受信バッファと、

前記シリアル転送バスを介して送信されるデータを受信し前記送受信バッファに格納すると共に、前記送受信バッファに格納されたデータを送信するデータ送受信ユニットと、

前記データ送受信ユニットによる転送データの送受信が行われている間に、前記CPUから前記送受信バッファへのアクセスが行われた場合に、当該アクセス発生 of フラグを格納するアクセスフラグレジスタとを有することを特徴とする集積回路装置。

【0081】

(付記13) 付記12において、

前記送受信マクロは、更に、

前記所定単位 of データ of 送信に応答して受信側デバイスから送信されるデータアクノリッジ信号を検出するアクノリッジ信号検出ユニットを有し、

前記データ送受信ユニットは、前記アクノリッジ信号検出ユニット手段が前記データアクノリッジ信号を検出することに応答して、前記CPUへの割込を発生することなく前記送受信バッファに格納されたデータを送信することを特徴とする集積回路装置。

【0082】

(付記14) 付記12において、

前記送受信マクロは、更に、

前記所定単位 of データ of 受信に応答して、送信側デバイスにデータアクノリッジ信号を送信するアクノリッジ信号生成ユニットを有し、

前記アクノリッジ信号生成ユニットは、受信可能なデータ単位数に達するまで、前記所定単位 of データ of 受信に応答して、前記CPUに割込を発生することなく前記データアクノリッジ信号を送信し、前記受信可能なデータ単位数に達した時に、前記データアクノリッジ信号を送信しないことを特徴とする集積回路装置。

【図面の簡単な説明】

【0083】

【図 1】 本実施の形態におけるシリアル転送バスを示す図である。

【図 2】 本実施の形態における集積回路装置であるマイクロコンピュータの全体構成図である。

【図 3】 本実施の形態における送受信マクロの構成図である。

【図 4】 シリアル転送バスのドライブ回路の一例を示す図である。

【図 5】 シリアル転送バスの信号波形図である。

【図 6】 第 1 の実施の形態を説明するタイミングチャート図である。

【図 7】 バスアービトレーション手続きを説明するための図である。

【図 8】 本実施の形態における送受信マクロの動作フローチャート図である。

【図 9】 本実施の形態におけるシーケンスチャート図である。

【図 10】 本実施の形態における別のシーケンスチャート図である。

【図 11】 本実施の形態における送受信マクロの一部の構成を示す図である。

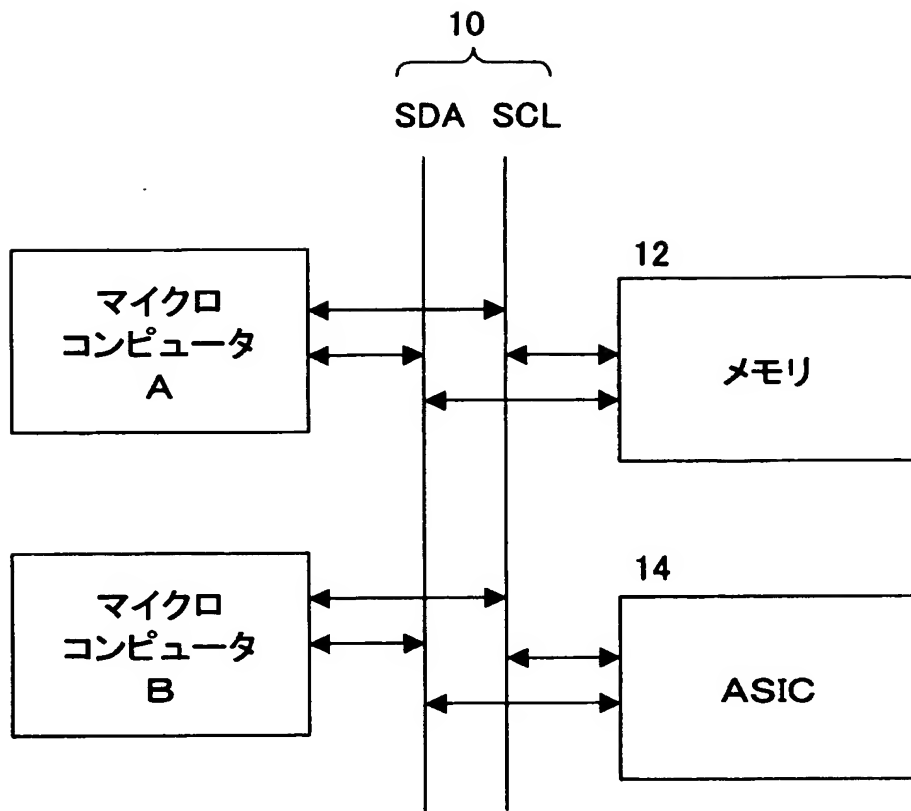
【図 12】 本実施の形態におけるシーケンスチャート図である。

【符号の説明】

【0084】

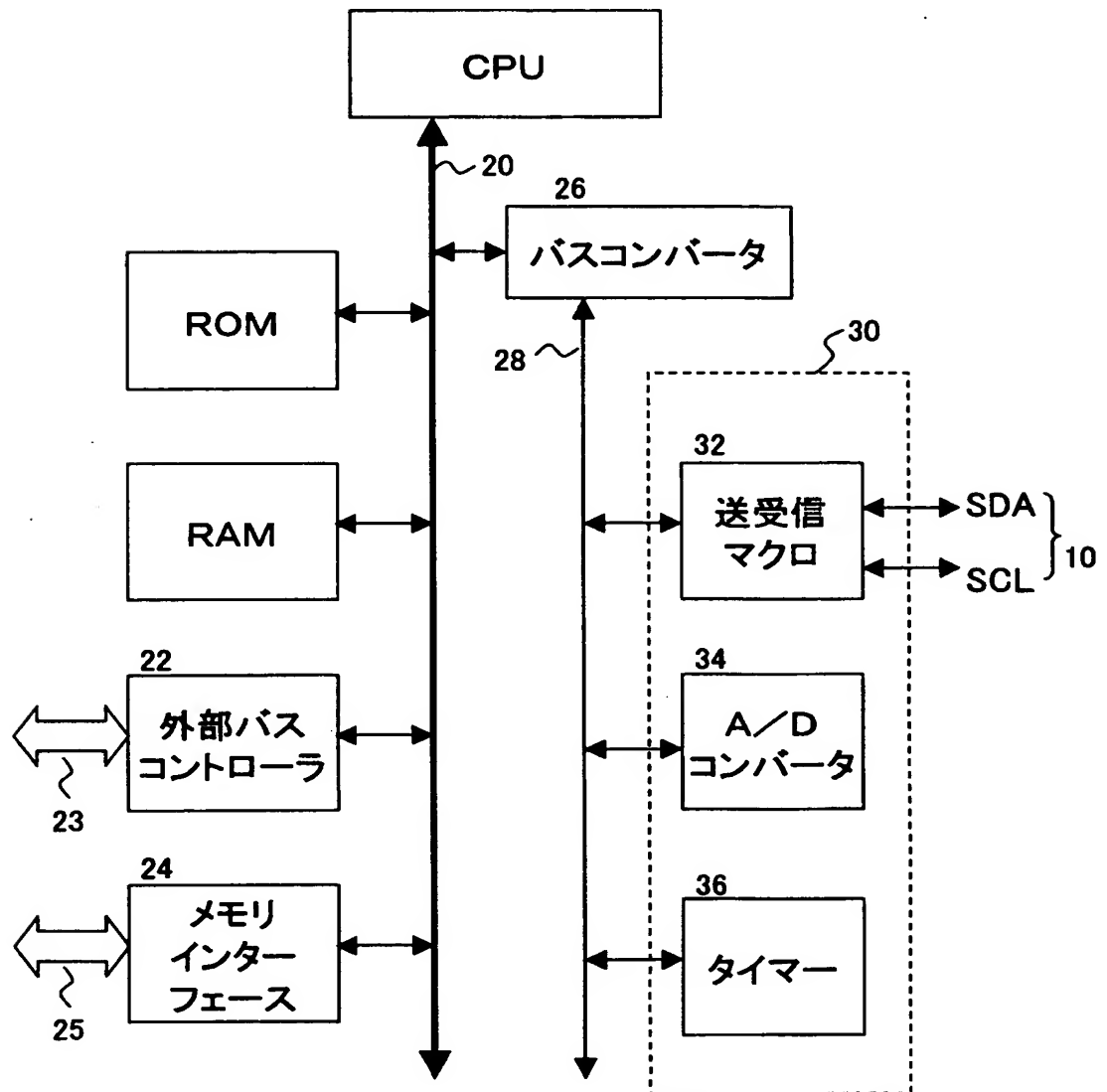
10：シリアル転送バス、32：送受信マクロ、46：送受信バッファ、  
48：データ送信・受信ユニット、50：アービトレーションロスト検出ユニット、  
52：アクノリッジ発生ユニット、54：アクノリッジ検出ユニット

【書類名】 図面  
【図 1】

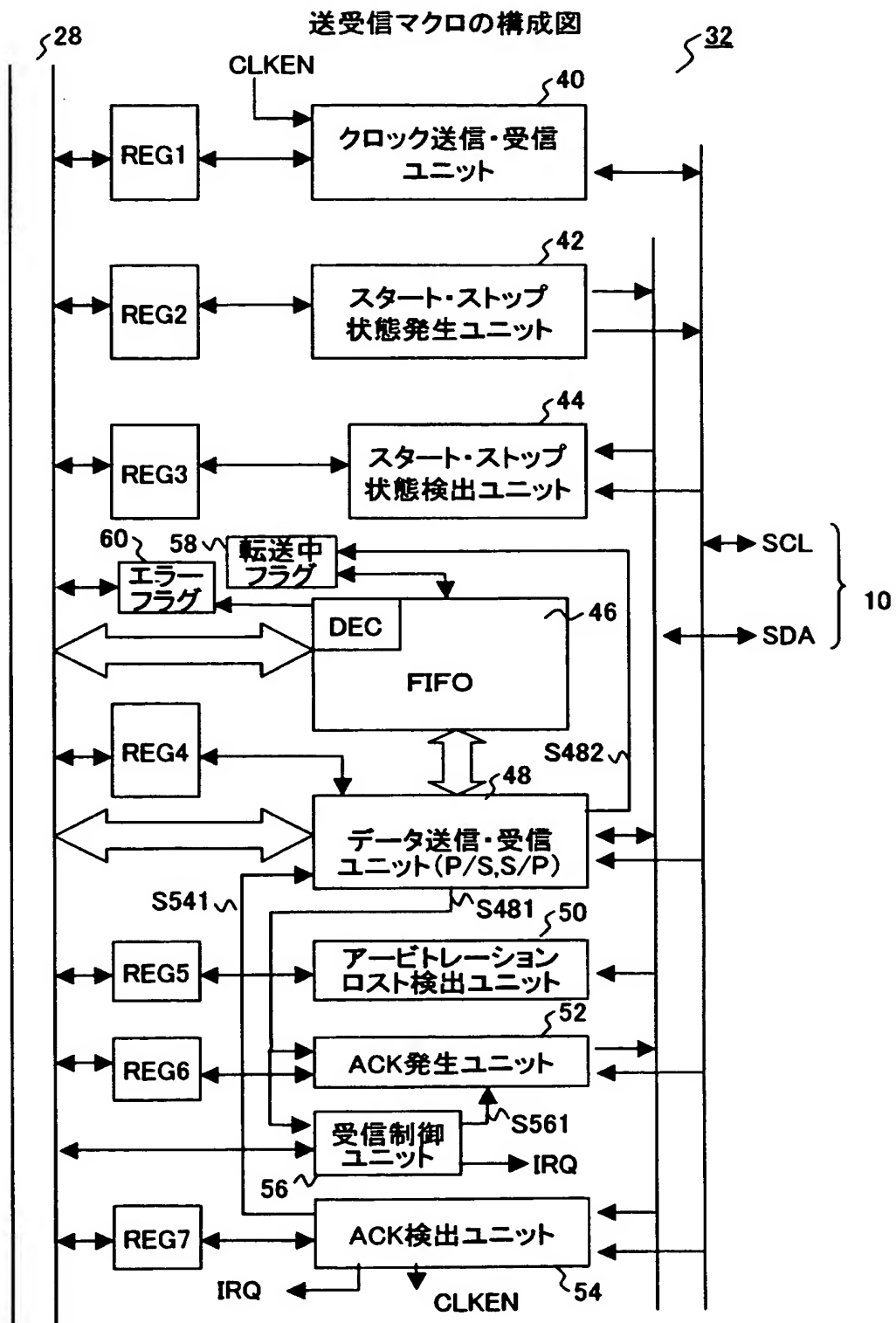


【図 2】

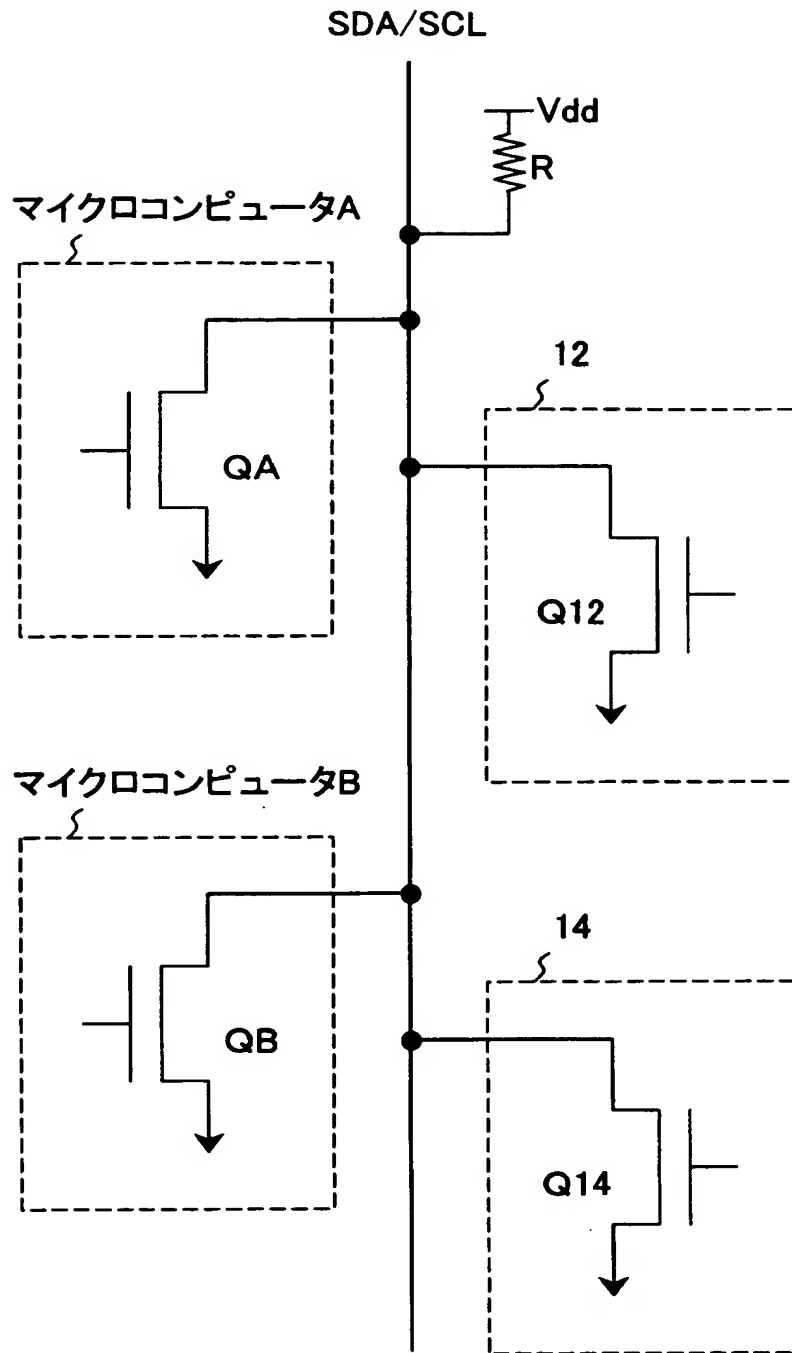
## 本実施の形態におけるマイクロコンピュータの全体構成図



【図 3】

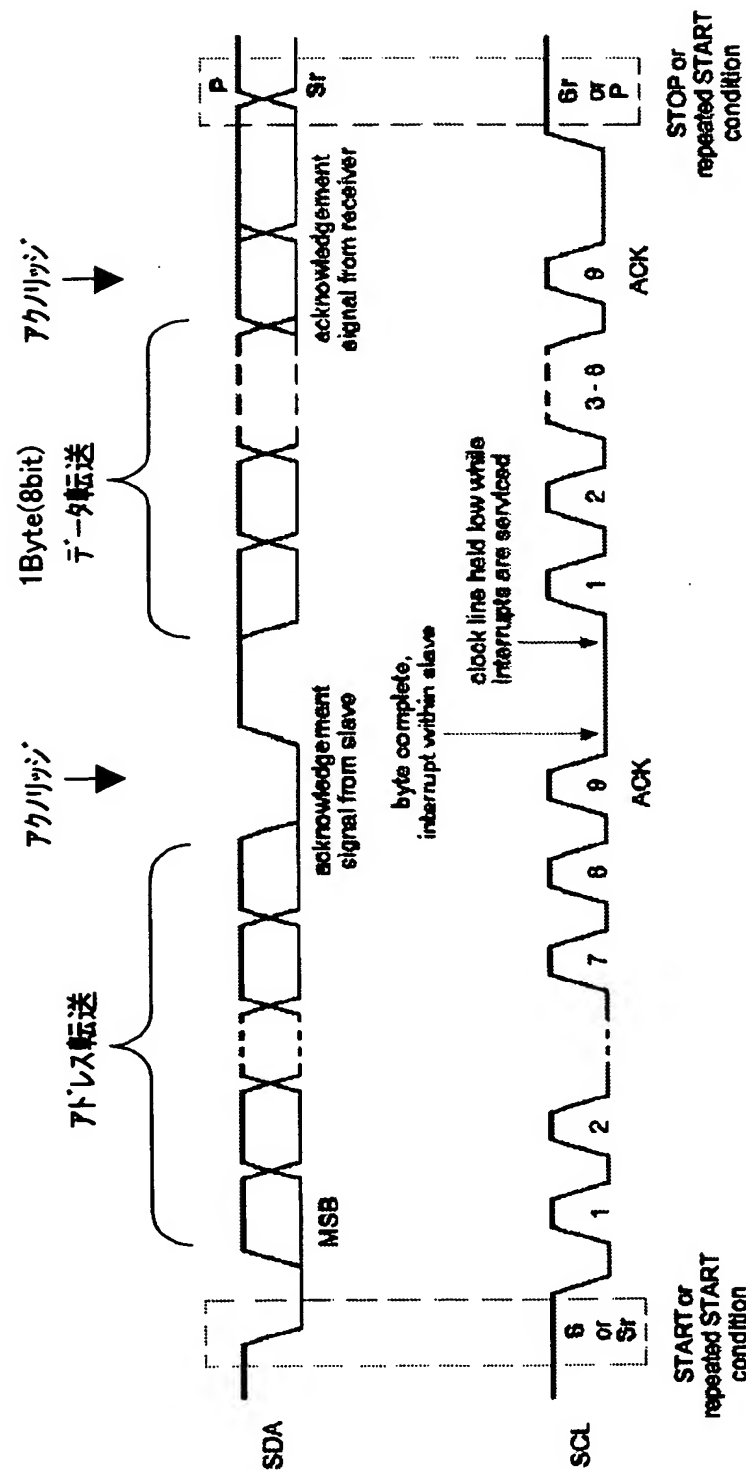


【図 4】

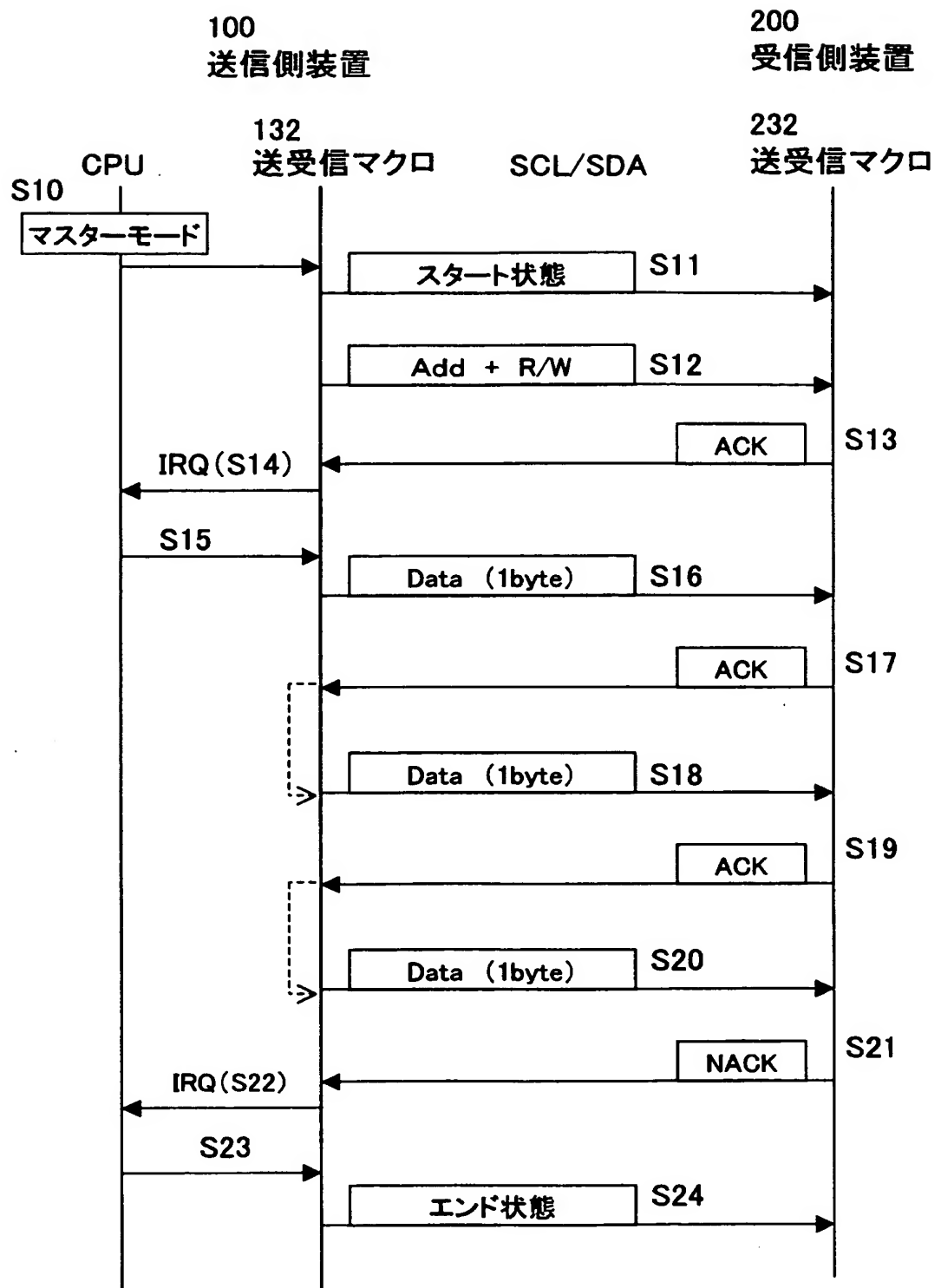




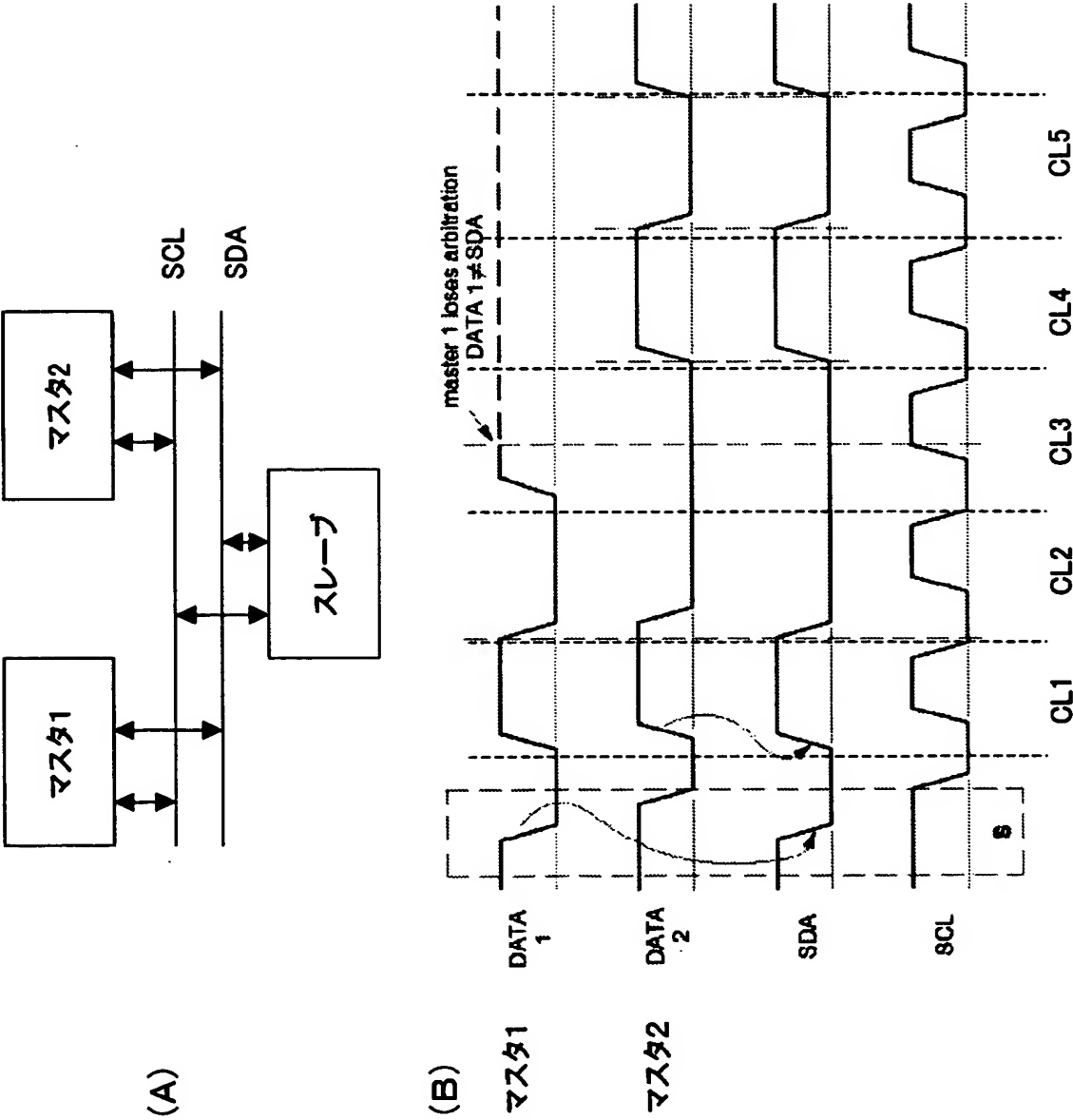
【図5】



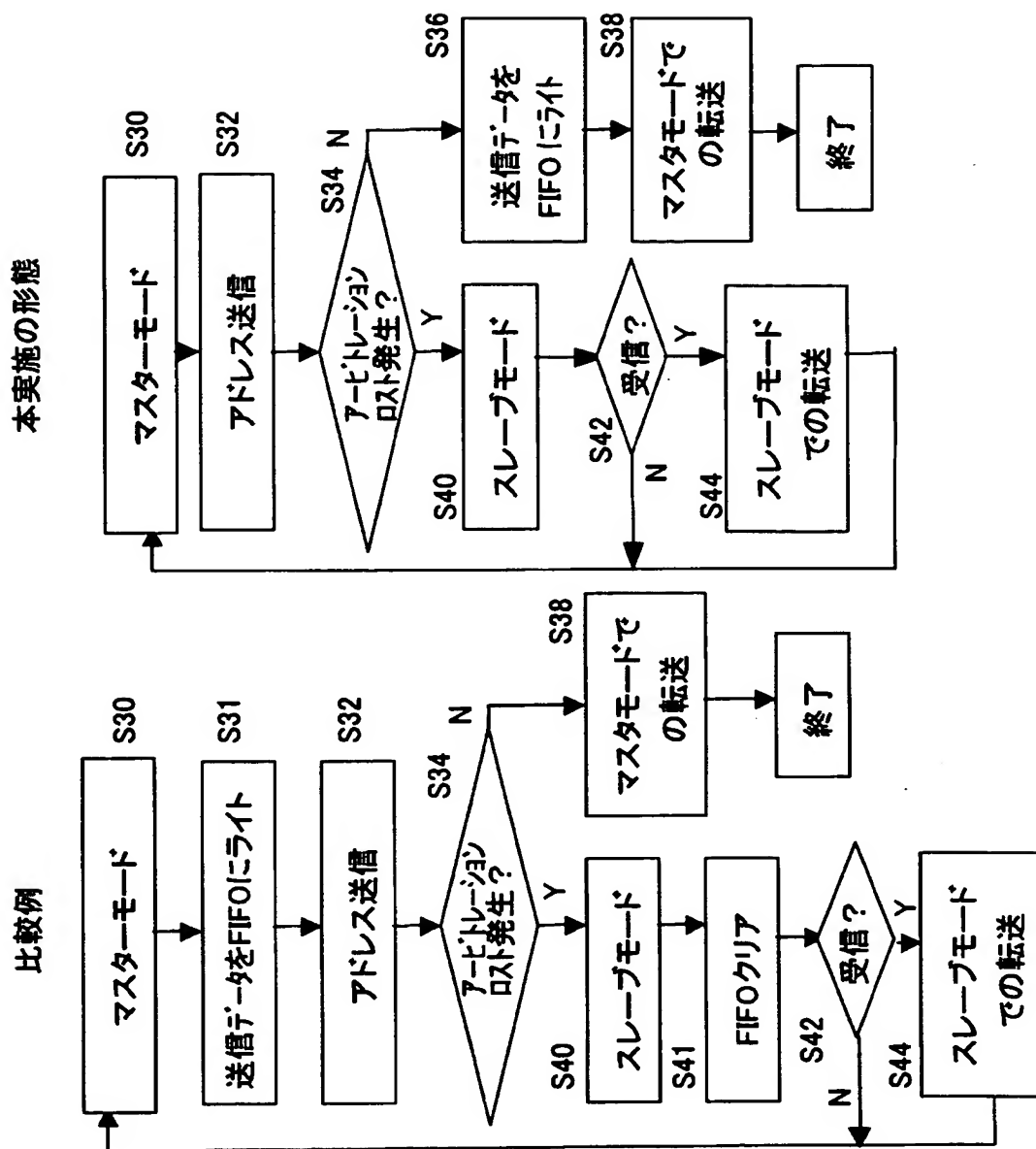
【図 6】



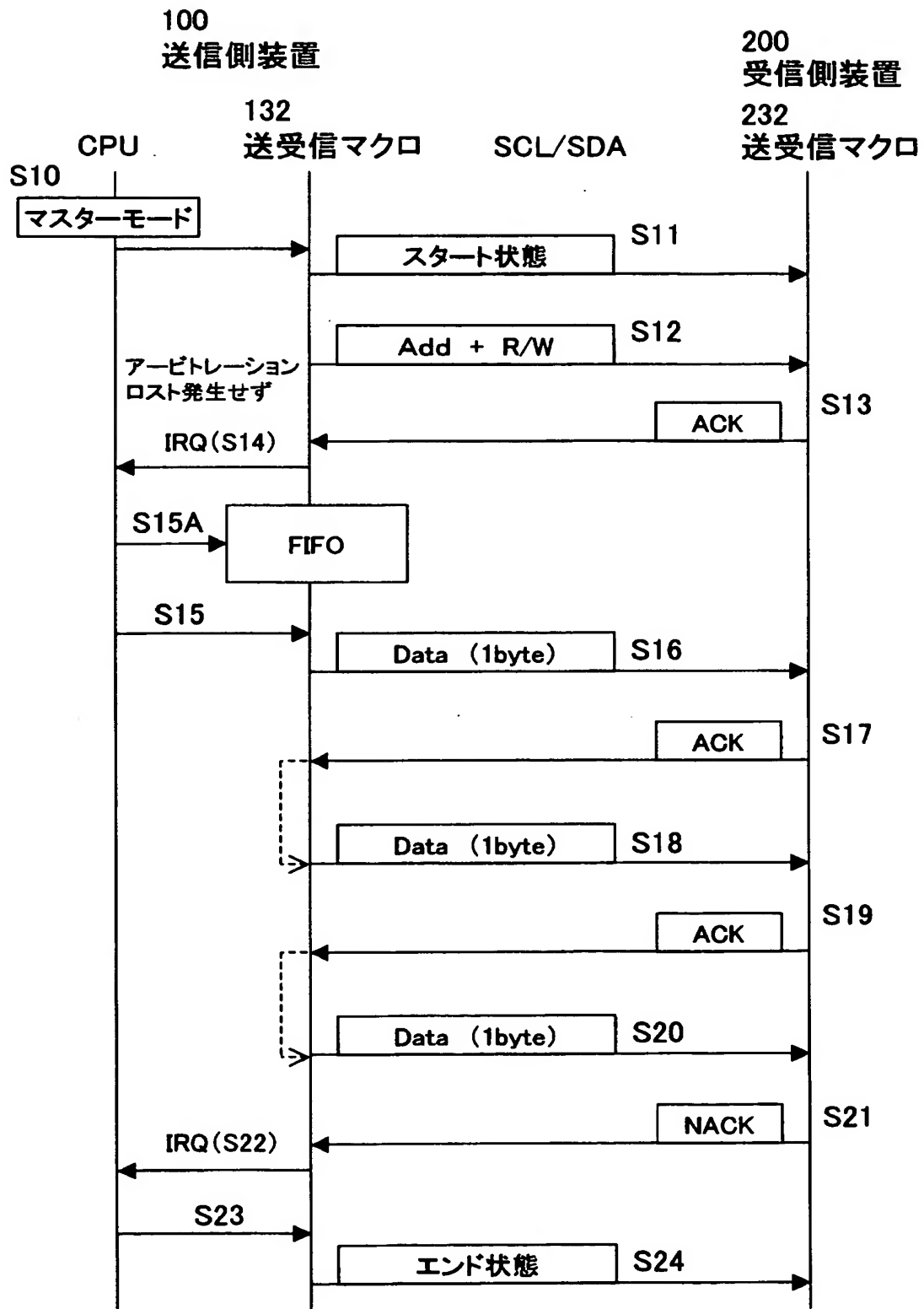
【図 7】



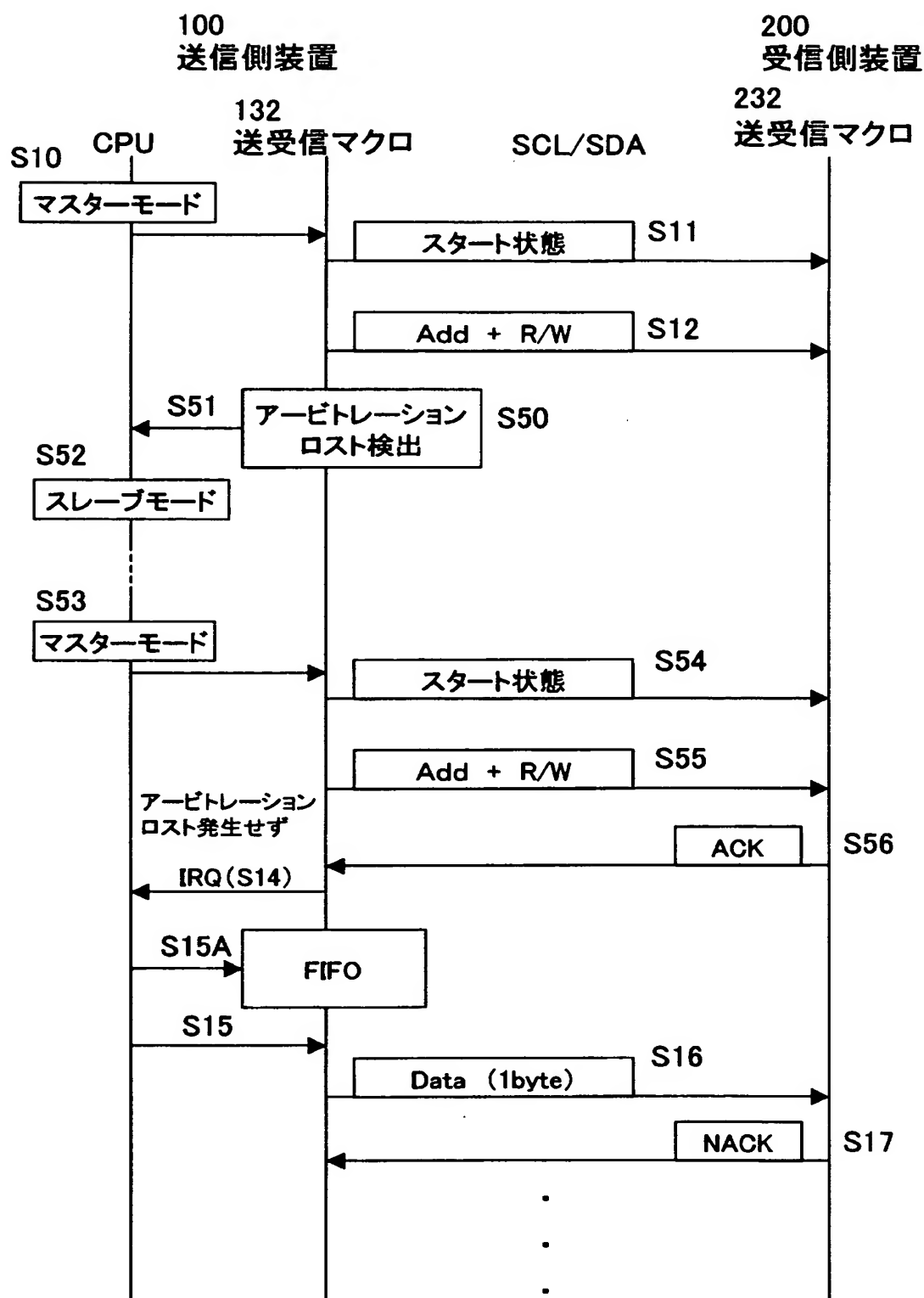
【図 8】



【図 9】

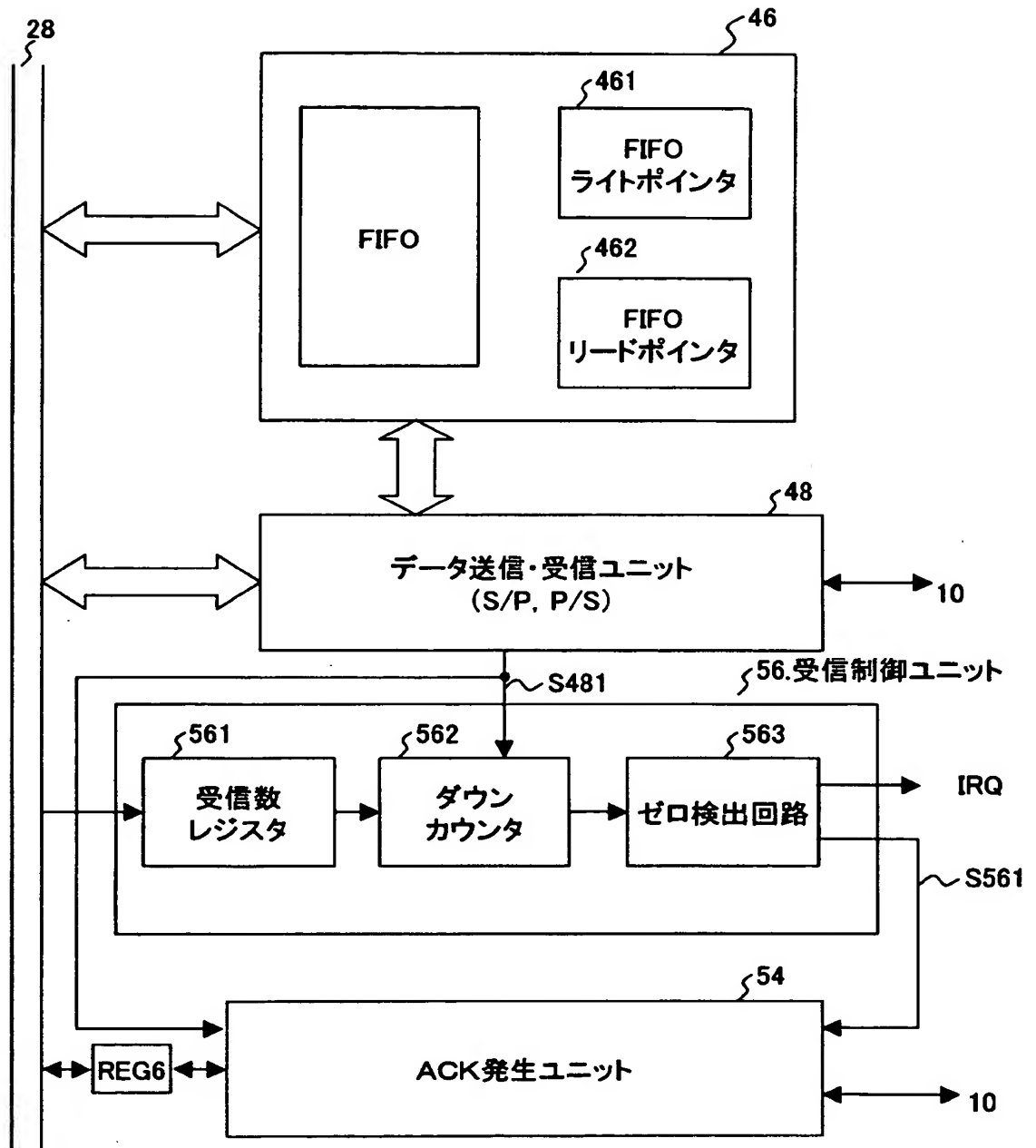


【図10】

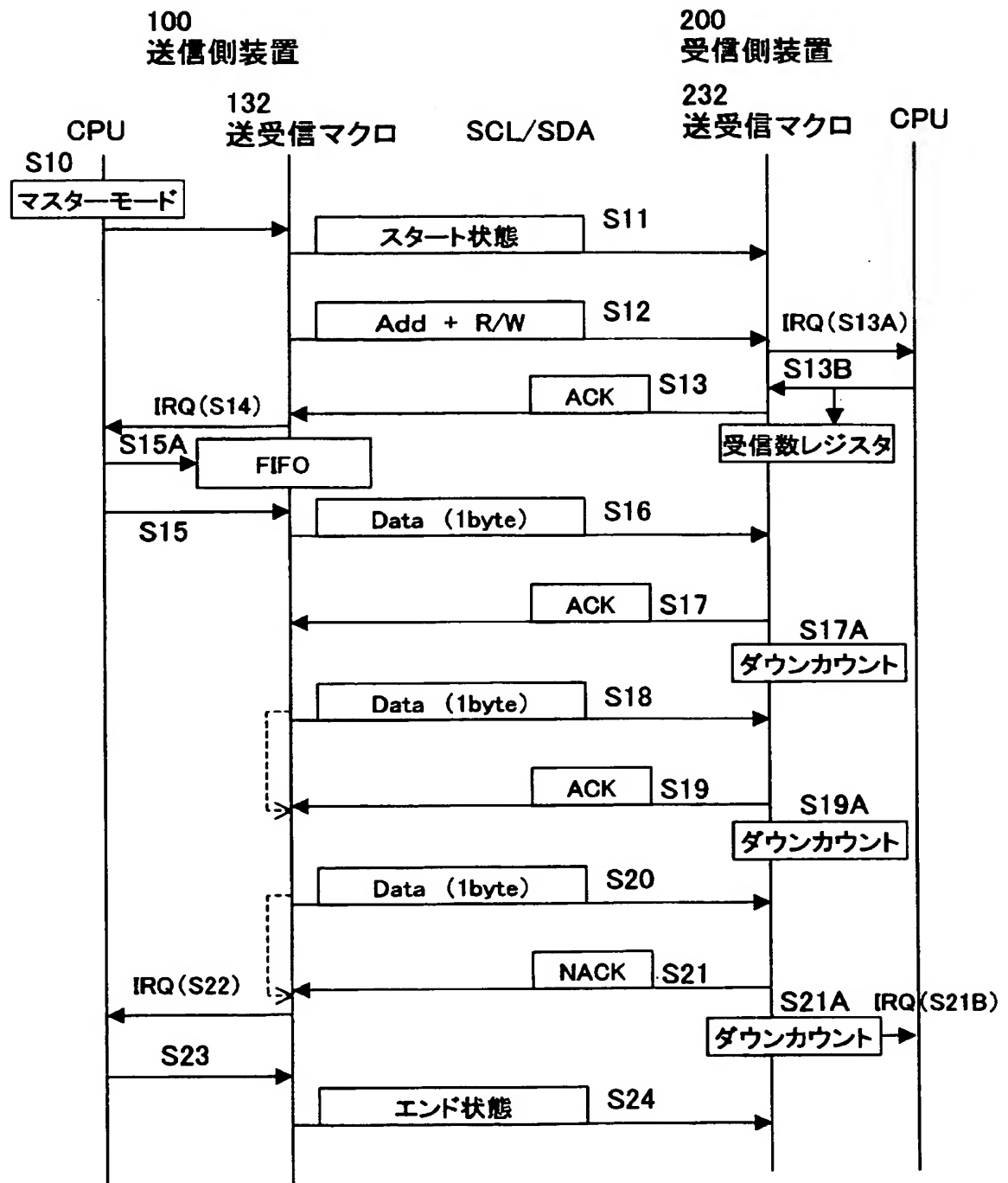


【図 11】

第3の実施の形態



【図 12】





**【書類名】 要約書****【要約】**

**【課題】** 送受信マクロにFIFOなどの送受信バッファを設け、CPUへの割込頻度を上げることなく、シリアル転送バスのデータ転送量を増大させる。

**【解決手段】** 集積回路装置は、所定の処理を実行するCPUと、シリアル転送バスを介して外部デバイスとアドレス及びデータをシリアル転送する送受信マクロをと有する。この送受信マクロは、CPUからアクセスされ、シリアル転送バスに対して送信または受信される複数単位のデータを格納する送受信バッファと、所定単位のデータの送信に応答して受信側デバイスから送信されるデータアクノリッジ信号を検出するアクノリッジ信号検出ユニットと、アクノリッジ信号検出ユニットによるデータアクノリッジ信号の検出に応答して、CPUへの割込を発生することなく送受信バッファに格納されたデータを送信するデータ送信ユニットとを有する。そして、アクノリッジ検出ユニットは、所定単位のデータの送信に対応して受信側デバイスから送信されるデータアクノリッジ信号を検出しない場合に、CPUに割込を発生する。これにより、CPUはデータ転送を終了させることができる。

**【選択図】** 図3

特願 2 0 0 3 - 3 6 3 0 3 8

出 願 人 履 歴 情 報

識別番号

[ 0 0 0 0 0 5 2 2 3 ]

1 . 変更年月日

1 9 9 6 年    3 月 2 6 日

[変更理由]

住所変更

住 所

神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号

氏 名

富士通株式会社